

タブーサーチによる FIR デジタルフィルタの 係数乗算ブロック最適化

松本 有弘[†] 佐々木孝雄[†] 豊嶋 久道[†]

[†] 神奈川大学工学部 〒221-8686 神奈川県横浜市神奈川区六角橋 3-27-1
E-mail: †{arihiro,sasaki,toyo}@tysm.ee.kanagawa-u.ac.jp

あらまし FIR デジタルフィルタの設計において、係数の量子化により特性は劣化するものの、乗算ブロックとして演算器を共有化させることにより計算量の削減が可能である。本研究では、グラフ表現された係数乗算ブロックに対してタブーサーチを用いることで、計算量、フィルタ特性のバランスを考慮した柔軟性のある最適化を可能とし、最適な係数の組合せを効率的に探索する方法を提案する。

キーワード タブーサーチ, FIR デジタルフィルタ, 乗算ブロック, グラフ表現

Optimization of Coefficient Multiplication Block for FIR Digital Filters Using Tabu Search

Arihiro MATSUMOTO[†], Takao SASAKI[†], and Hisamichi TOYOSHIMA[†]

[†] Faculty of Engineering, Kanagawa University Rokkakubashi 3-27-1, Kanagawa-ku, Yokohama-shi,
Kanagawa, 221-8686 Japan
E-mail: †{arihiro,sasaki,toyo}@tysm.ee.kanagawa-u.ac.jp

Abstract In a design of FIR digital filters, though quantization of coefficients deteriorates filter performance, it is possible to reduce complexity by implementing multipliers as a multiplication block using common subexpressions. In this research, we propose a method for searching the optimum coefficients as a multiplication block efficiently using Tabu Search algorithm. This method provides the flexible optimization in consideration of both complexity and filter performance.

Key words tabu search, FIR digital filter, multiplication block, graph representation

1. まえがき

FIR デジタルフィルタは、完全な直線位相特性を容易に実現でき、デジタル信号処理に関連した多くの分野で重要な役割を果たすものである。FIR デジタルフィルタの設計においては、Remez 法などの手法を用いる事で仕様を満たした等リプルフィルタの設計が可能である。しかし実際の回路設計においては、このフィルタ係数を演算回路のビット長に合わせて量子化せねばならず、それによりフィルタ特性の劣化を招いてしまう。このフィルタ設計問題については古くから研究が行われており、特性を改善させる手法が多数提案されている [1] [2] [3]。

一方で、FIR デジタルフィルタは、その回路中に複数の乗算器が用いられているが、その乗算部を乗算ブロックとして、加減算器とシフトに置き換えるアルゴリズムが幾つか提案されている [4] [5] [6] [7]。この手法を用いて演算器を共有化させることにより、計算量の削減が可能である。

FIR デジタルフィルタの設計では、このように性能面での特性と回路規模面での計算量の両面を考慮して設計する必要がある。特性と計算量を考慮した設計法として、それぞれの係数を 2 の巾乗 (Signed Power-of-Two, SPT) 項の集合で表現する事により、組合せ最適化問題に帰着させ、シミュレーテッドアニーリング (Simulated Annealing, SA)、遺伝的アルゴリズム (Genetic Algorithm, GA)、タブーサーチ (Tabu Search, TS) などのアルゴリズムを用いたフィルタ係数設計法が提案されている [8] [9] [10]。

また、計算量削減の観点から考えると、乗算ブロック化されたフィルタ係数に対して最適化を行う方が好ましいが、演算器共有自身も最適化問題となり、その解空間は膨大になるため、これに対して組合せ最適化アルゴリズムを適用している例は少ない。その一つとして GA を用いた文献 [11] などが提案されているものの、グラフ構造に基づいてフィルタ特性を向上させる場合の最適化には局所的な係数変化も必要とされるため、大域

的探索を得意とする GA が適しているとは一概には言い難い。さらに、GA や SA は、一般的に最適化に長い計算時間を必要とする欠点もある。

このような背景から本研究では、組合せ最適化アルゴリズムの中でも計算時間が比較的短く、局所探索を得意とする TS に着目し、フィルタ係数をグラフ構造に基づいて変化させ、フィルタの計算量を削減させるとともに、量子化により劣化したフィルタ特性を向上させる方法を提案する。また、評価式の重みを調整する事で、計算量あるいはフィルタ特性のバランスを考慮した最適化が可能となる。

具体的な手法としては、まずフィルタ係数の乗算部をブロック化し、その乗算ブロックを非巡回有向グラフで表現する。さらにそのグラフ表現された係数乗算ブロックに対して TS を用い、グラフ中のそれぞれのノードをグラフ中で取り得る近傍値へ遷移させることで、フィルタ係数の最適化を行う。また、その際のノードの変化方法として、グラフ構造の大域的な変化と局所的変化の両面からの最適化が行えるよう、タブーリストを利用して変化させる係数の数を制御するとともに、TS を並列化させる事によって、より効率的な探索をおこなっている。

2. FIR デジタルフィルタにおける乗算ブロックの計算量削減

FIR デジタルフィルタの伝達関数は次式のように表せる。

$$H(z) = \sum_{n=0}^{N-1} C_n z^{-n} \quad (1)$$

また、図 1 のように転置型構成とすることにより、乗算ブロックの部分が複数の定数乗算回路と見なせる。

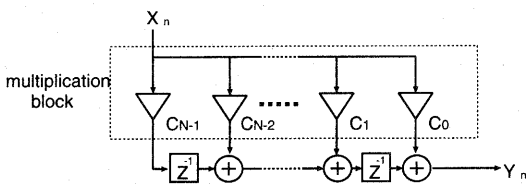


図 1 転置型 FIR フィルタの構成

この乗算ブロックは、それぞれ加減算とシフトで構成される各乗算の加算器を互いに共有することにより計算量を削減することが可能である。加算とシフトで構成された乗算ブロックは非巡回有向グラフとして表現される。その際、与えられたフィルタ係数に対して、非巡回有向グラフを設計するアルゴリズムの一つである BHM アルゴリズム (Modified Bull and Horrocks Algorithm, BHM) [5] を用いる事により、乗算ブロックをグラフ構造として構成する事が可能である。BHM はシフトを基礎として分岐と加減算を繰り返して係数を形成していくもので、新たに形成された加算結果をシフトさせていき、シフトチェーンに格納する。シフトチェーン中の値は他の係数へ利用できる事から、計算量の削減が可能である。例として係数セット {5, -11, 19, 25} による乗算ブロックを考えた場合、図 2

のように係数を生成するのに必要な要素の一部分を共有することにより、乗算器単位の回路に比べ、加算器とシフト数を減少させる事が可能である。なお、フィルタ係数の符号情報は、0 の場合正值を、1 の場合負値を出力することを表す。

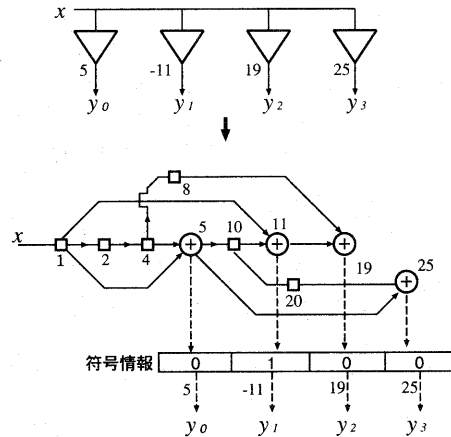


図 2 乗算ブロックの合成 (BHM)

3. タブーサーチによる FIR デジタルフィルタの係数乗算ブロック設計

本研究では、乗算ブロックを有向グラフ化した際のグラフ構造に基づき、回路の計算量を削減するとともにフィルタ性能を向上させる事を目的とする。

これは組合せ最適化問題の一つであるが、設計可能な係数の組合せは定数セットのビット長やフィルタ次数によっては膨大な範囲となり、最適化が困難な問題である。そこで本研究ではその最適化手法として、メタ戦略の一つである TS を用いる。

TS は局所探索法に改良を加えた最適化手法であり、解の近傍という考えに基づき、その近傍への遷移の際にタブーリストと呼ばれる記憶を用いることで解の循環を避けるとともに解の多様化の役割を果たし、効率的な探索をおこなうアルゴリズムである。TS は他の組合せ最適化アルゴリズムに比べ、より細かなパラメータ設定が可能であり、この手法をブロック化係数に適用することで、グラフ構造に沿った細かな最適化が可能である。

3.1 近傍

TS では現在の解に対して次の解への変化を行う際、その近傍の解を生成し、その近傍解への遷移という考えを利用する。それぞれの近傍解に対し算出された評価値を基に、タブーリストにない最良解へと遷移する。

本研究では、グラフ構造中で係数が出力されるノードに対しての近傍を考える。

BHM では、グラフを生成していく過程で用いられた数とそのビットシフト値が図 3 のようなシフトチェーンとして保存される。例えば、{5, 17, 18, 20} がグラフ中で係数として生成されている状態での近傍を考える。

ノードの変化例として、図3のノード値17(C_1)のように、2数の加算によるノードに対する近傍は、シフトチェーン中にある数から1数を選択、あるいは2数の加減算により生成し、選移させるノード値に近いものから順に近傍とする。図4の変化例では、ノード値17が16へと変化する事により、加算ノードがシフトノードへと変換されている。また、その影響を受けノード値18(C_2)も同時に17へと変化している。

また、図3のノード値20(C_3)のように、係数がシフトノードにより生成されているものは、2数の加減算をおこなうと加算器数が増加してしまう。そのため、シフトノードの場合には、シフトチェーン中からの1数の選択、あるいはシフトの基数(この場合では5)の加算ノードに対する2数の加減算により近傍を生成する。図5の変化例では、基数の加算ノード値5(C_0)を6へと変化させる事により、加算器数を増加させる事なくノード値20を24へと変化させている。

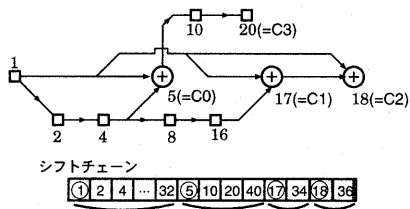


図3 シフトチェーン例

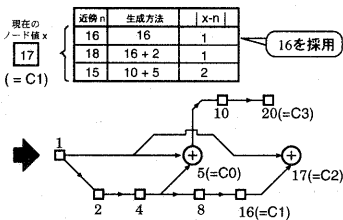


図4 加算ノードの変化例

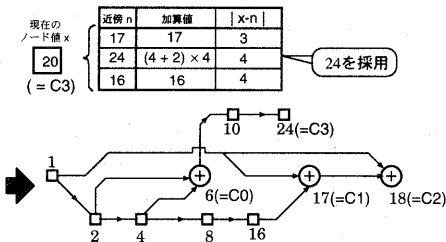


図5 シフトノードの変化例

また、特に0付近の係数の変化を考えた場合、符号の変化が重要となる場合があるため、上記の手法では適した近傍を生成できない。そこで、0付近の係数についてはグラフ構造に基づいた近傍とはせず、例外として図6のように符号情報の変化を中心とした近傍を考える。

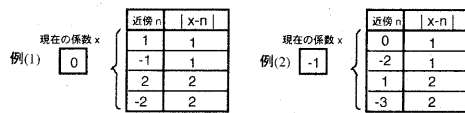


図6 符号変化も含めた近傍例

3.2 タブーリスト

近傍への遷移をする際、遷移した履歴などの情報をタブーリストとして記憶しておくことで解の循環を防止するとともに探索の多様化をおこなう。本研究では3種類の記憶をタブーリストとして用いる。

short-memory …… それぞれの係数に対し用意し、係数の変化履歴を記憶する。これにより解の循環を防ぐとともに未探索の領域への探索の方向づけを行う。*short-memory*の長さは解空間により異なるが、本研究では各係数に対し1~3とし、ビット長やフィルタ次数に応じて変化するものとする。

long-memory1 …… 各係数に対し用意し、各係数が変化した回数を記憶する。これにより変化がしやすい係数、しにくい係数を分類し、変化させる係数を選択する際に利用する。初期の*long-memory1*内は全ての係数に対して1とし、係数が変化した際に更新するものとする。

long-memory2 …… 今までに得られた最良解よりも良い解が見つからない回数を記憶する。一回以上優れた最良解が発見されない場合は、探索を終了する。

3.3 タブーサーチの適用

以下に今回用いたTSによる最適化の流れを、またそのフローチャートを図7、図8に示す。初期解としては、Remez法により設計したフィルタ係数を有限ビット長で丸め、その絶対値を整数化した上で有向グラフとして表現したものを用いる。

(1) 変化させる係数を選択する。係数を選択する際には*long-memory1*を利用し、係数が変化した回数に比例した割合で解を選択する手法(ルーレット方式)を用いる。これにより、変化した回数が多い係数は、選択される確率が高くなるようにする。

また、フィルタ係数は他の係数とのバランスが特性に大きく影響を与えているため、係数単体で変化させるよりも複数で変化させた方が最適な場合が多い。そこで本研究では、複数の係数を同時に選択するようにしている。この際、図9のようにグラフ表現した際にグラフ内部に位置し、変化させる事で多くの係数に影響するノードと、グラフの末端部に位置し、他の係数に影響がないノードが同時に選択される事がないよう制御する。

さらに、*long-memory1*の情報により、探索の初期は複数の係数を選択する頻度が高いが、探索が進むにしたがって徐々に選択する係数の数が少なくなるように制御する事により、大域的探索から局所的探索へと移行をおこなっている。

(2) 変化させる係数に対応するノードに対する近傍を生成する。近傍の数は選択された係数の数によって異なり、3~64とする。

(3) それぞれの近傍に対して評価値を算出し、最も評価値の高い近傍を新しい値として採用する。その際、最良解を基に再構成された係数が*short-memory*に含まれている場合、その近

傍値はタブーとなり、採用しない事とする。ただし、その最良解が今までに得られた最良解よりも優れたものである場合、タブーリストに含まれていてもその解を採用する事とする(願望水準)。

(4) 以上の TS を同じ係数に対し変化させる係数を変えて並列的におこなう(図 8)。それぞれの TS の結果の評価値を比較し、最も評価値が高い解を最良解として採用する。その際、新しく変化させた係数を *short-memory* へ追加するとともに、その係数に対応する *long-memory1* の変化回数を更新する。また、得られた解が今までに得られた最良解よりも良い解となった場合には *long-memory2* を初期化する。逆に良い解が見つからなかった場合には 1 を加える。

(5) *long-memory2* を参照し、探索終了条件を満たしていなければ、(1)に戻る。

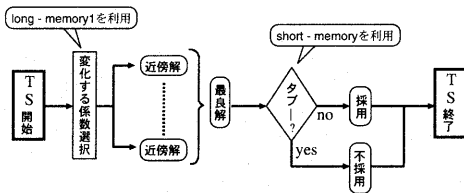


図 7 タブーサーチの流れ

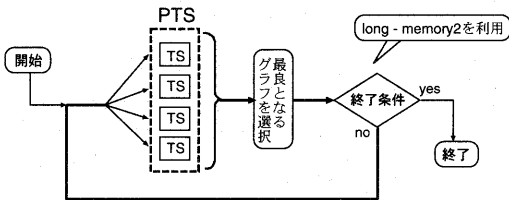
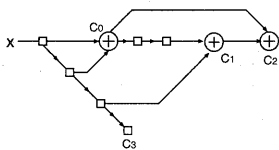


図 8 最適化の流れ



$C_0 \cdot C_1$ 変化することで他の係数にも影響する
 $C_2 \cdot C_3$ 変化することで他の係数に影響しない

図 9 各ノードの他の係数への影響

3.4 評価値

近傍を評価する際に用いたフィルタ性能に対する評価値 P_i を次式で表す。

$$P_i = \frac{1}{W_p \cdot E_{pi} + W_s \cdot E_{si}} + W_a \cdot N_{ai}$$

E_{pi} : 通過域最大リプル誤差

E_{si} : 阻止域最大リプル誤差

N_{ai} : 初期解からの加算器数の削減数

W_p, W_s, W_a : E_{pi}, E_{si}, N_{ai} に対する重み

ここで P_i は i 番目の近傍に対する評価値であり、設計されたフィルタの通過域および阻止域の最大リプル誤差が小さいもの、あるいは加算器の削減数が多いものほど評価値が高くなる。

また、それぞれの重みを変化させることで、通過域・阻止域の誤差あるいは加算器数の削減に重点を置いた最適化が可能となる。

4. シミュレーション

4.1 最適化例 1

フィルタの仕様を表 1 に、TS のパラメータ設定を表 2 に示す。

表 1 フィルタ仕様

| フィルタの種類 | Low Pass Filter |
|---------------------|-----------------|
| フィルタ次数 | 43 |
| ω_p [πrad/s] | 0.4 |
| ω_s [πrad/s] | 0.5 |
| ビット長 [bit] | 10 |

表 2 TS パラメータ

| <i>short-memory</i> | 各係数に対し 2 |
|---------------------|---------------------------|
| W_p | 1.5 |
| W_s | 1.0 |
| W_a | 1.0 |
| 並列数 | 4 |
| 探索終了条件 | <i>long-memory2</i> ≥ 100 |

従来法との係数比較を表 3 に示し、それらの係数の絶対値を整数化し、グラフ構造化したものを図 10、図 11 に示す。グラフ中での下線の引いてあるノードは、そのノードから係数を出力している事を表している。また、その係数により得られたフィルタ特性および加算器数の比較を表 4、図 12、図 13 に示す。なお、比較対象は文献 [3] の PTS によってフィルタ特性が最適化された係数に対して BHM を適用したものをを用いた。

これらの結果より従来法に比べ、加算器数の削減が行いながらフィルタ性能の最適化ができていくことがわかる。また、従来法のように特性のみを最適化し、その後グラフ構造化したものと比べ、本手法はグラフ構造を基本とした最適化をおこなっているため、計算量を念頭に置いた最適化が可能である事が分かる。

また、この場合の計算時間は約 5 分 (SUN Ultra-SPARC3, 750MHz 使用) であった。

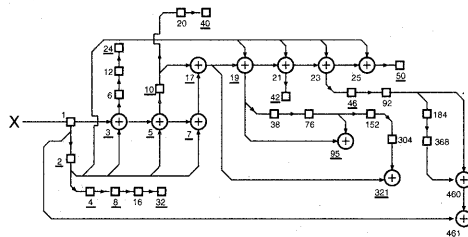


図 10 最適化前の乗算ブロック (PTS)

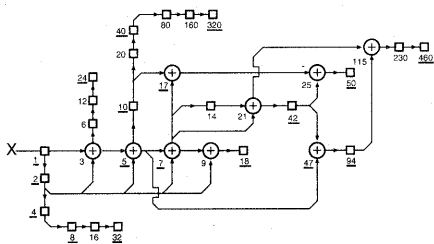


図 11 最適化後の乗算ブロック

表 3 フィルタ係数比較 ($\times 2^{-10}$)

| Coefficients | rounded remez | PTS | 提案法 |
|-------------------|---------------|-----|-----|
| $C_0 = C_{42}$ | -5 | -4 | -4 |
| $C_1 = C_{41}$ | 1 | 0 | 1 |
| $C_2 = C_{40}$ | 6 | 6 | 5 |
| $C_3 = C_{39}$ | 3 | 2 | 2 |
| $C_4 = C_{38}$ | -6 | -6 | -7 |
| $C_5 = C_{37}$ | -5 | -5 | -5 |
| $C_6 = C_{36}$ | 8 | 8 | 8 |
| $C_7 = C_{35}$ | 11 | 10 | 10 |
| $C_8 = C_{34}$ | -7 | -7 | -7 |
| $C_9 = C_{33}$ | -17 | -17 | -17 |
| $C_{10} = C_{32}$ | 3 | 3 | 4 |
| $C_{11} = C_{31}$ | 25 | 24 | 24 |
| $C_{12} = C_{30}$ | 5 | 4 | 4 |
| $C_{13} = C_{29}$ | -32 | -32 | -32 |
| $C_{14} = C_{28}$ | -18 | -19 | -18 |
| $C_{15} = C_{27}$ | 40 | 40 | 40 |
| $C_{16} = C_{26}$ | 43 | 42 | 42 |
| $C_{17} = C_{25}$ | -46 | -46 | -47 |
| $C_{18} = C_{24}$ | -94 | -95 | -94 |
| $C_{19} = C_{23}$ | 50 | 50 | 50 |
| $C_{20} = C_{22}$ | 321 | 321 | 320 |
| C_{21} | 461 | 461 | 460 |

表 4 結果比較

| | 加算器数 | 阻止域 [dB] | 通過域 [dB] | $W_p E_p + W_s E_s$ |
|---------------|------|----------|---------------|---------------------|
| REMEZ | - | -42.5332 | ± 0.06490 | 0.01863 |
| rounded REMEZ | 13 | -39.6174 | ± 0.09392 | 0.02676 |
| PTS | 12 | -41.1211 | ± 0.07369 | 0.02157 |
| 本手法 | 9 | -41.2473 | ± 0.07605 | 0.02173 |

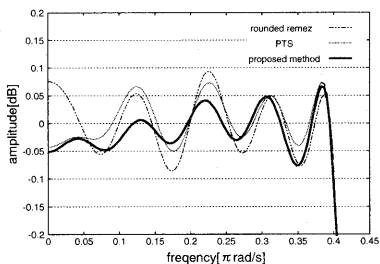


図 12 通過域特性

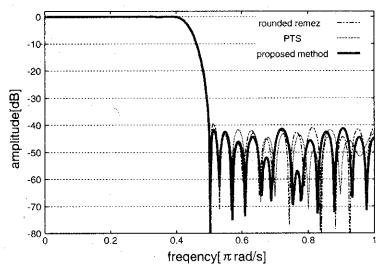


図 13 阻止域特性

4.2 最適化例 2

フィルタの仕様を表 1 に、TS のパラメータ設定を表 2 に示し、 W_p を変化させて最適化をおこなった時の結果の違いを比較した。

表 5 フィルタ仕様

| フィルタの種類 | Low Pass Filter |
|---------------------|-----------------|
| フィルタ次数 | 65 |
| ω_p [πrad/s] | 0.25 |
| ω_s [πrad/s] | 0.3125 |
| ビット長 [bit] | 10 |

表 6 TS パラメータ

| short - memory | 各係数に対し 2 |
|----------------|--------------------------|
| W_p | 1 |
| W_s | 1 |
| 並列数 | 4 |
| 探索終了条件 | long - memory ≥ 100 |

表 7 結果比較

| | 加算器数 | 阻止域 [dB] | 通過域 [dB] | $W_p E_p + W_s E_s$ |
|-------------------|------|----------|--------------|---------------------|
| REMEZ | - | -39.9389 | ± 0.0877 | 0.02012 |
| rounded REMEZ | 15 | -36.2894 | ± 0.1179 | 0.02900 |
| PTS | 16 | -38.5451 | ± 0.1036 | 0.02368 |
| 本手法 ($W_a=0.2$) | 12 | -38.7657 | ± 0.1058 | 0.02364 |
| 本手法 ($W_a=3$) | 10 | -37.9336 | ± 0.1144 | 0.02578 |

表 7 より、評価値中の W_a の重みを軽くした場合には特性を重視した最適化を、逆に W_a の重みを重くした場合には加算器数を重視した最適化が行われている事がわかる。

5. まとめ

本研究では FIR デジタルフィルタの設計において、タブーサーチを用いてフィルタ係数を変化させ乗算ブロックを最適化することで、フィルタ特性を改善するとともに回路の計算量を削減する方法を提案した。グラフ構造の加算部に対し、並列的に TS を用いるとともに、タブーリストを積極的に探索方法へ応用する事により、効率的に直線位相特性を維持したままフィルタ特性を向上させることができた。

また、本研究においては、GA や SA ではなく、局所的探索を得意とするタブーサーチを用いた事で、グラフ構造に基づいた効率の良い最適化を可能とした。また、パラメータの調整により、回路の計算量とフィルタの特性のトレードオフの関係において、両面のバランスを考慮した最適化を行う事ができた。

今後は GA 等のアルゴリズムと組み合わせ、より効率の良い最適化手法の検討が課題として挙げられる。また、加算段数なども考慮し、遅延時間の短縮も可能な最適化手法の検討も今後の課題である。

文 献

- [1] D.M.Kodek, "Design of Optimal Finite Wordlength FIR Digital Filters Using Integer Programming Techniques", *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. ASSP-28, pp.304-308, 1980.
- [2] D.M.Kodek and K.Steiglitz, "Finite-length word-length

- tradeoffs in FIR digital filters design", *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. ASSP-28, pp.739-744, 1980.
- [3] D. Karaboga, D.H.Horrocks, N.Karaboga, A.Kalinli, "Designing Digital FIR filters Using Tabu Search Algorithm", *IEEE ISCAS*, pp.2236-2239, 1997.
- [4] Bull and Horrocks, "Primitive operator digital filters", *IEE Proc. G*, 138, (3), pp.401-412, 1991.
- [5] A.G.Dempster and M.D.Macleod, "Constant integer multiplication using minimum adders", *IEE Proc. Circuits Devices Syst.*, vol.141, no.5, pp.407-413, Oct. 1994.
- [6] A.G.Dempster and M.D.Macleod, "Use of minimum adder multiplier blocks in FIR digital filters", *IEEE Trans. Circuits Syst.*, vol.42, no.9, pp.569-577, Sept. 1995.
- [7] M.Potkonjak, M.B.Srivastava, and A. P. Chandrakasan, "Multiple constant multiplication: Efficient and versatile framework and algorithms for exploring common subexpression elimination", *IEEE Trans. Computer-Aided Des. Integrated Circuits & Syst.*, vol.15, no.2, pp.151-165, Feb. 1996.
- [8] N.Benvenuto, M.Marchesi, A.Uncini, "Applications of Simulated Annealing for the Design of Special Digital Filters", *IEEE Trans. Signal Proc.*, vol.40, no.2, pp.323-332, Feb. 1992.
- [9] P.Gentili, F.Piazza, A.Uncini, "Efficient Genetic Algorithm Design for Power-of-two FIR Filters", *Proc. of Int. Conf. on Acoust., Speech, Signal Proc. 1995*, pp.1268-1271, 1995.
- [10] S.Traferro, F.Capparelli, F.Piazza, A.Uncini, "Efficient Allocation of Power-of-two Terms in FIR Digital Filter Design Using Tabu Search", *Proc. of 1999 IEEE Int. Symp. on Circuits and Systems*, Vol.3, pp.411-414, Jul. 1999.
- [11] D.W.Redmill, D.R.Bull, E.Dagless, "Genetic synthesis of reduced complexity filters and filter banks using primitive operator directed graphs", *IEE Proc. Circuits Devices Syst.*, vol.147, no.5, pp.303-310, Oct. 2000.